Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 31 (2014) 860 – 868

Procedia Computer Science

2nd International Conference on Information Technology and Quantitative Management, ITQM 2014

Using Formal Concept Analysis for Finding the Closest Relatives Among a Group of Organisms

Alena Lihonosova[a,]*, Alexandra Kaminskaya[a,b]

[a]*National Research University Higher School of Economics, 20 Myasnitskaya St., Moscow, 101000, Russia*
[b]*Yandex, 16, Leo Tolstoy St., Moscow, 119021, Russia*

Abstract

The paper presents a study on comparing different organisms, which requires their DNA sequences. If one considers a sample of DNA regions, an interesting result can be obtained. By using formal concept analysis a procedure that allows to determine the strongest family among different organisms is proposed. The methodology is explained in this paper.

Keywords: Formal concept analysis; DNA sequences; Comparing genomes

1. Introduction

Comparison of genome sequences and chromosomes is one of the main tasks of bioinformatics and biological research. The difference between organisms is the difference between their genomes. Hence, by comparison of two or more genomes, scientists can understand the evolutionary relationships between them, because the amount of differences should indicate how recently these genomes shared a common ancestor, and vice versa, the majority of identical regions in DNA of different species show the closest relatives. Nowadays, sequence alignment is fundamental starting point for comparing genomes. There is pairwise alignment, which was proposed by Saul B. Needleman, and Christian D. Wunsch[1], or multiple alignments for comparing strings, which was considered, for

* Corresponding author.
E-mail address: alihonosova@gmail.com

doi:10.1016/j.procs.2014.05.337

instance, by Hohl M, Kurtz S, Ohlebusch E[2]. The idea is to arrange blocks of DNA sequences in such a way that the number of identical blocks is maximized throughout the columns. The distance between genomes is the minimum number of changes sufficient to transform one sequence to another[3]. Clustering methods can be used for finding close relatives and building phylogenetic trees. For example, there is Unweighted Pair Group Method with Arithmetic Mean (UPGMA)[4], whose idea is to build a distance matrix for genomes and then divide them into clusters. This paper is going to present a method for detecting the closest relatives in any group of organisms that based on finding only identical regions of genomes and do not consider distances between them. Common identical regions in genomes of any group can indicate characteristic features of this group. The more these regions organisms shared, the more common features they have, therefore, the closer they are. There is an assumption that the closest relatives are those, whose genes share the majority of common unique DNA regions, in other words, they have the majority of common features that makes them different from the others.

Formal concept analysis was used in bioinformatics previously for many purposes. There is the work where the two FCA-based methods for mining numerical data in the context of gene expression data analysis are proposed and compared[5]; the approach based on formal concept analysis to classify and search relevant bioinformatics data sources for a given user query[6] was proposed; it was used for extraction of co-expressed genes, namely genes with similar expression pattern[7]; this method was used for finding disease similarity[8]. However, there are no known results for finding phylogenetic relatives among a group of organisms by using FCA.

The structure of the paper is the following. First, the description of dataset and its statistical analysis will be provided. Second, the graphical visualization of data will be proposed. Third, the concept lattice will be presented. Finally, the conclusion, and the weak and strong points will be discussed.

Nomenclature

$K = (G, M, I)$	formal context
G	set of objects
M	set of attributes
$I \subseteq M \times G$	binary relation that shows which objects possess which attributes
A	subset of objects
B	subset of attributes

2. Dataset

A DNA sequence is a code written in only four letters, called A, C, T and G. The meaning of a DNA code is in a sequence of these letters. Similarly, the meaning of a word is in a sequence of alphabet letters. The analyzed dataset is from freely available online genome database for vertebrates and other eukaryotic species[9]. Five following organisms are considered: human, chimpanzee, horse, dog and mouse. The first chromosomes of each species are analyzed. Each of these chromosomes consists of about two hundred billion letters.

2.1. Statistical analysis

Before the DNA comparison, let us look at how these chromosomes are arranged. The stacked bar chart in fig. 1 illustrates proportions of base pairs (four different letters) in the given chromosomes. Four different colors indicate four different letters. The histogram in fig. 2 shows the shares of dimers (two successive letters). There are sixteen different dimers. Finally, codons (sixty four different three successive letters) are considered, and stacked line chart with markers in fig. 3 shows their proportions in five chromosomes. A particular marker indicates the proportion for particular codon stacked with the other ones, calculated before. The lines which connect particular markers are used to show differences in proportions of codons in five sequences. In the light of space limitation, the legends in fig.2 and fig.3 are not represented fully. To sum up, as we can see from the charts, five given chromosomes have almost

the same proportions of the letters and the subwords. However, they are not identical; therefore, there should be subwords of some length in each sequence which are unique for a group of one or more species.

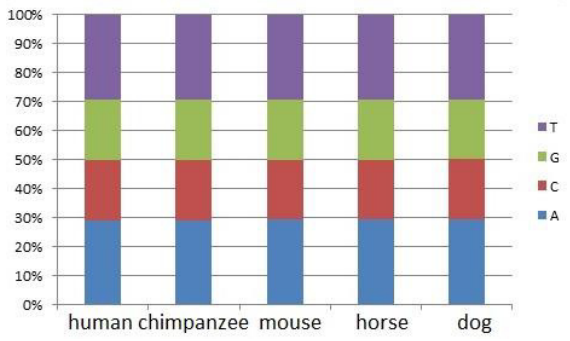

Fig. 1. Stacked proportions of base pairs in each chromosome.

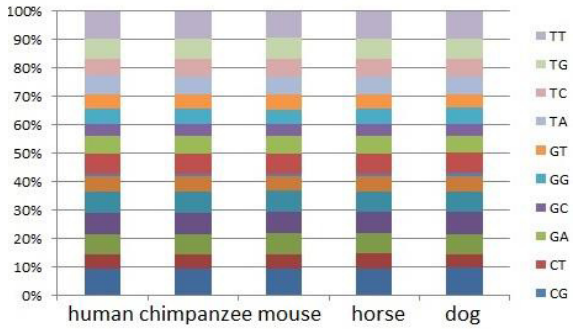

Fig. 2. Stacked proportions of dimers in each chromosome.

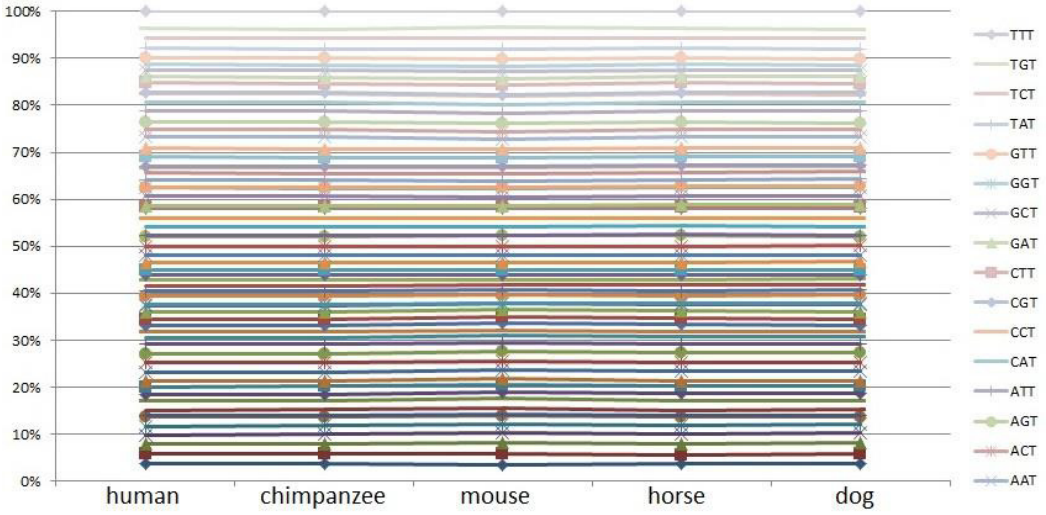

Fig. 3. Stacked proportions of codons in each chromosome.

2.2. Sample of DNA regions

It was obtained experimentally, launching the program (see code in Appendix A) for different lengths, that substrings of any chromosome which length is fifteen letters can be used to determine similarity between subgroups, because shorter words are often found in each chromosome and longer ones occur only in those chromosomes from which they were obtained. The sample of such substrings is made by selecting one hundred of random substrings from each chromosome. As a result, the sample contains five hundred of substrings. Then the number of occurrences of these substrings in each sequence is counted. The program for counting can be found in Appendix A. Fig. 4 (a) shows the fragment of the table with the result. The first column is the list of substrings, where the first row indicates species, the cell (i,j) represents the number of occurrences of the subword i in the chromosome j. Not all of these substrings allow us to make a sensible decision about similarity between species. For example, it is difficult to compare five species by using the first substring from the table in fig. 4 (a), but the highlighted substrings in fig. 4 (b) indicate similarity within a group of species. Only genes which occur in any subgroup the same time and do not occur in the others in the group of five species are considered. So there should be at most two numbers of occurrences in any row. There is an exception for large figures, which in comparison with the others in particular row are almost the same, as it can be seen in row 465 and 469 in fig.4 (a). Thereby, the meaningless subwords are removed from the sample.

a

	A	B	C	D	E	F
446		dog	horse	mouse	human	chimpanzee
448	TTTTGAGCCTATGTT	1	2	1	3	4
449	AGCCTGCGTAACAGA	0	0	0	3	3
450	ACACAAGAACAAGCA	0	0	0	1	1
451	CACACACTGTGCACG	0	0	0	1	1
452	TTACAGGCACACGTC	0	0	0	7	6
453	GGTCACCTGGCCAGT	1	0	0	2	2
454	TTTGAGACATGTAAT	1	0	1	1	1
455	ATGTTTCACAATTTT	3	6	1	3	3
456	CTTATTTGATGCTTT	1	1	1	3	4
457	TAATCTTAAACTTCC	0	1	2	2	2
458	AATGTTTATTTAAAA	2	7	7	14	16
459	GGATTTAATGTGAAG	0	0	0	1	1
460	GACTCTGGGTGGTGT	0	0	0	1	1
461	TTTAGATCCCTGGGT	0	0	0	1	1
462	CGCACATTGCGGATC	0	0	0	1	1
463	TTTCCAGTGGTTTAT	0	2	0	1	2
464	AGGTCTTTAGTGACA	0	0	1	1	1
465	ACTCCTGACCTCAGG	1	0	0	5080	4806
466	TATATTTTGTAATCA	1	2	1	3	3
467	CTCATATATGTACAT	1	0	3	2	1
468	TGAAGGAATAGAAAA	3	3	1	5	6
469	GCACTCCAGCCTGGG	0	2	0	16319	15158
470	CTCCCGAAGGTGCAC	0	0	0	1	1
471	GTTGGGAAAATTGAA	1	2	0	1	1
472	TTTCTGTGGGTCAAG	2	4	0	4	4
473	GTGGGGAGATGGGAT	2	0	1	0	1
474	CCATGCCTGGCCATC	0	0	0	6	7
475	GTATCTATTTGTCTT	0	1	0	1	2
476	GAAAACAGTATAAAA	0	0	1	3	2

b

	A	B	C	D	E	F
446		dog	horse	mouse	human	chimpanzee
448	TTTTGAGCCTATGTT	1	2	1	3	4
449	AGCCTGCGTAACAGA	0	0	0	3	3
450	ACACAAGAACAAGCA	0	0	0	1	1
451	CACACACTGTGCACG	0	0	0	1	1
452	TTACAGGCACACGTC	0	0	0	7	6
453	GGTCACCTGGCCAGT	1	0	0	2	2
454	TTTGAGACATGTAAT	1	0	1	1	1
455	ATGTTTCACAATTTT	3	6	1	3	3
456	CTTATTTGATGCTTT	1	1	1	3	4
457	TAATCTTAAACTTCC	0	1	2	2	2
458	AATGTTTATTTAAAA	2	7	7	14	16
459	GGATTTAATGTGAAG	0	0	0	1	1
460	GACTCTGGGTGGTGT	0	0	0	1	1
461	TTTAGATCCCTGGGT	0	0	0	1	1
462	CGCACATTGCGGATC	0	0	0	1	1
463	TTTCCAGTGGTTTAT	0	2	0	1	2
464	AGGTCTTTAGTGACA	0	0	1	1	1
465	ACTCCTGACCTCAGG	1	0	0	5080	4806
466	TATATTTTGTAATCA	1	2	1	3	3
467	CTCATATATGTACAT	1	0	3	2	1
468	TGAAGGAATAGAAAA	3	3	1	5	6
469	GCACTCCAGCCTGGG	0	2	0	16319	15158
470	CTCCCGAAGGTGCAC	0	0	0	1	1
471	GTTGGGAAAATTGAA	1	2	0	1	1
472	TTTCTGTGGGTCAAG	2	4	0	4	4
473	GTGGGGAGATGGGAT	2	0	1	0	1
474	CCATGCCTGGCCATC	0	0	0	6	7
475	GTATCTATTTGTCTT	0	1	0	1	2
476	GAAAACAGTATAAAA	0	0	1	3	2

Fig. 4. (a) The number of occurrences of subwords from the sample in each chromosome; (b) The useful data are highlighted

As a result, 284 subwords that are useful for determining similarity within subgroups of chromosomes are obtained. Then the binary matrix is constructed by replacing the numbers of occurrences by one, because only similarities are considered.

3. Graphical visualization

Now a formal context can be defined as K = (G, M, I), where G is a set of objects, that consists of five chromosomes of the given five organisms, M is a set of attributes, that consists of the sample of 284 subwords. I is the relations, where (g,m) in I if chromosome g consist subword m. The following operators (1) and (2) are defined for subsets A, B:

$$A' = \{m \in M | \forall g \in A\ (g, m) \in I\} \quad (1)$$

$$B' = \{g \in G | \forall m \in B\ (g, m) \in I\} \quad (2)$$

A pair (A, B) is a formal concept if $A' = B$ and $B' = A$. In other words, the formal concept indicate relationships among objects which have the same set of attributes and attributes which are associated with the same set of objects.

3.1. Bipartite graph

The formal context can be visualized as a bipartite graph $K = < W, I >$, where $W = G \cup M$. Relation I is an edge that connect g and m if $(g, m) \in I$. As it can be seen from the graph in fig.5 there are twenty eight formal concepts, which are obtained by grouping attributes which correspond to the same set of objects. Twenty three subsets of attributes are in the bottom of the graph and five unique sets of attributes for each chromosome, which are not illustrated, form the concepts.

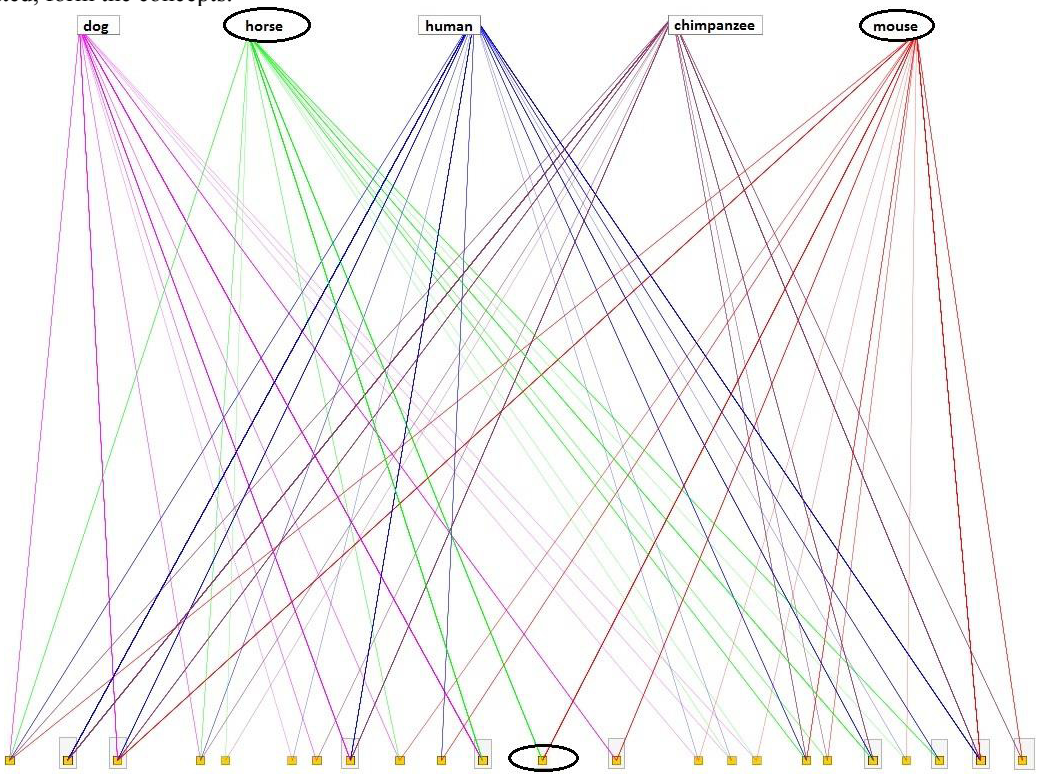

Fig. 5. Illustration of the context with five objects and 284 grouped attributes with the example of formal concept.

3.2. Graph of pairwise relations

In order to show relations between organisms we can define a graph $G = < V, E >$ that illustrates the connections between organisms, where V is a set of all species, and E are edges that connect two nodes by common gene regions.

Weights indicate the number of common genes. As we can see from the graph in fig. 6, there are much more similarities between human and chimpanzee; the next similar pairs are mouse and chimpanzee, and mouse and human. The graph of pairwise relations has only ten relations between objects. That is why it loses information about the relationships in the data that are captured by formal concepts.

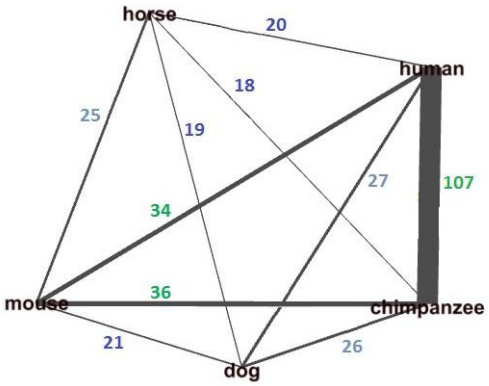

Fig. 6. Graph of pairwise relations between objects.

4. Concept lattice

The lattice in fig.7 is computed using the Concept Explorer. It represents concepts which can be difficult to see in the bipartite graph. The different levels of lattice represent different number of objects in the concepts. There is an assumption that a set of genes from any concept represents characteristic features of the particular group of organisms. So, we can define the strongest family as the group that has the majority of common unique genes from the sample. According to fig.7 the biggest set of common genes associated with the group that consists of human and chimpanzee. The second strongest family is the union of horse and dog, while in fig.6 the weight of their connection edge is one of the smallest ones. This is because their chromosomes share the common genes from the sample which the others chromosomes do not have. Therefore, they have the second biggest set of common characteristic features that makes them different from the others. For example, the body structures of horse and dog are similar but differ from the structure of human, chimpanzee and mouse, so they are phenotypically related. The third strong relative according the concept lattice is the union of chimpanzee, human and mouse. The graph of pairwise relations indicate the pairs of mouse and human, and mouse and chimpanzee as the second strong relatives. As we can see from the lattice, these groups have quite a few common unique genes but the next level shows that they share more by their union.

5. Conclusions

In this paper the formal concepts, which correspond to intersections of genes from the sample, are constructed. This method is sensible if each DNA region hold particular information about an organism. Twenty eight concepts, which represent different families, are found. The results can be described as the clusters in fig.8 (a). Also the unexpected strong family of horse and dog that the graph of pairwise relations defines as the weakest connection, was obtained. The clustering of the result of pairwise comparison by using neighborhood method is shown in fig. 8 (b). The main advantage of formal concept analysis is that it allows identifying the relations between the groups of organisms rather than the pairwise relationships. The weak point is that only the sample of substrings is considered. Also there is the personal selection of meaningful substrings from the sample that is ambiguous. Moreover, the positions of each subsequence in a chromosome are not considered. Overall, these results present a truly new

approach for finding the closest relatives, because in comparison to the state of the art in biology, only short identical fragments of a sequence are considered, while the known methods that are used nowadays are based on the whole sequence alignment. The obtained results haven't been consulted with domain experts yet. In addition, this method can be successfully applied to other similar analytical tasks in the field where the comparison of DNA sequences are used. For example, it can be applied to classification of plants. In this task the strongest relatives will indicate a particular class.

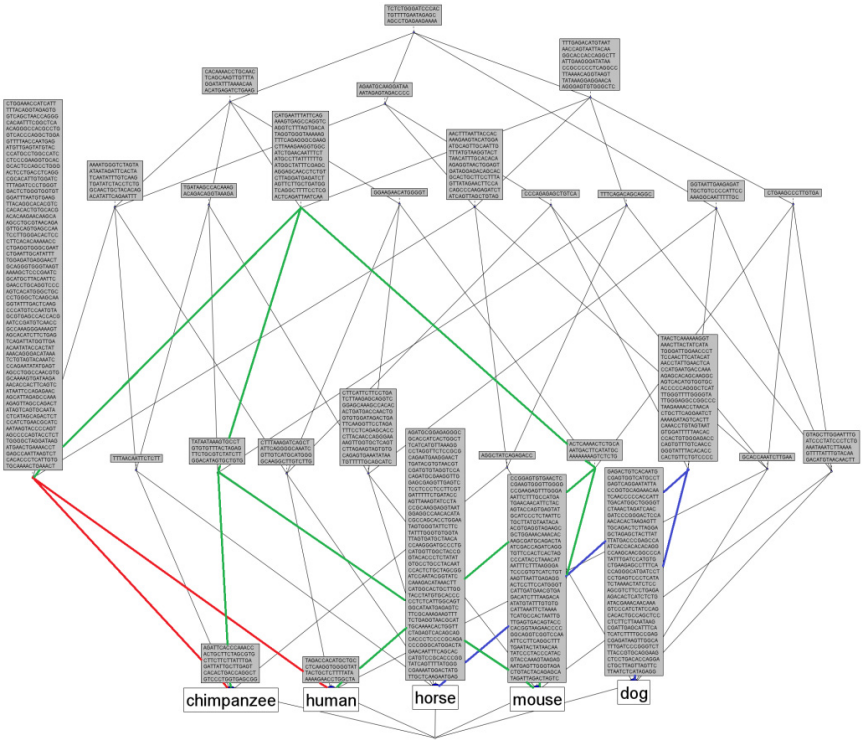

Fig. 7. Concept lattice

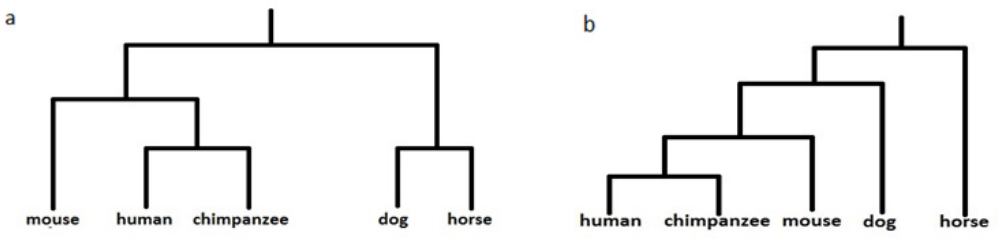

Fig. 8. (a) Clustering of the result of FCA; (b) Clustering of the results of pairwise comparing

Appendix A.

The program code for generating dataset is written in python. There is an example for generating a set of random subwords from the chromosome of chimpanzee and counting the numbers of their occurrences in other chromosomes.

```
from Bio.Seq import Seq
from Bio import SeqIO

#loading data
fasta_sequences1 = SeqIO.parse(open('D:\human.fa'),'fasta')
fasta_sequences2 = SeqIO.parse(open('D:\chimpanzee.fa'),'fasta')
fasta_sequences3 = SeqIO.parse(open('D:\mouse.fa'),'fasta')
fasta_sequences4 = SeqIO.parse(open('D:\horse.fa'),'fasta')
fasta_sequences5 = SeqIO.parse(open('D:\dog.fa'),'fasta')

#getting the sequences of letters from fasta format
for fasta in fasta_sequences1:
   name, sequence_human = fasta.id, fasta.seq.tostring();
for fasta in fasta_sequences2:
   name, sequence_chimpanzee = fasta.id, fasta.seq.tostring();
for fasta in fasta_sequences3:
   name, sequence_mouse = fasta.id, fasta.seq.tostring();
for fasta in fasta_sequences4:
   name, sequence_horse = fasta.id, fasta.seq.tostring();
for fasta in fasta_sequences5:
   name, sequence_dog = fasta.id, fasta.seq.tostring();

#generating random subwords from the human chromosome
import random
from random import randint
n = []     #array with the number of occurrences in each chromosome
gen = []   #array of random subwords
k = 100    #the number of subwords
for i in range(k):
   n.append([])

#filling the array n
for i in range(k):
   m = random.randrange(1,(len(sequence_chimpanzee)-15)+1)
   gen.append(sequence_chimpanzee[m:m+15]);
   n[i].append(sequence_dog.count(gen[i]));
   n[i].append(sequence_horse.count(gen[i]))
   n[i].append(sequence_mouse.count(gen[i]));
   n[i].append(sequence_human.count(gen[i]));
   n[i].append(sequence_chimpanzee.count(gen[i]));

#export to excel file
import xlwt
from xlwt import *
workbook = xlwt.Workbook()
sheet = workbook.add_sheet("chimpanzee.xls")
for i in range(len(gen)):
   sheet.write(i,0,gen[i])
```

```
    for j in range(len(n[i])):
        sheet.write(i,j+1,n[i][j])
workbook.save('D:\chimpanzee.xls')
```

References

1. Saul B. Needleman, Christian D. Wunsch. *Journal of Molecular Biology* 48(3): 443-453, 1970.
2. M. Hohl, S Kurtz, E Ohlebusch, Efficient multiple genome aligment, *Bioinformatics*, 18(SI):S312-S320, 2002.
3. R. A.Wagner, M. J. Fischer, The string-to-string correction problem, *Journal of the ACM*: 168-173, 1974.
4. I. Gronau, S. Moran, Optimal Implementations of UPGMA and Other Common Clustering Algorithms, *Information Processing Letters* 104(6): 205-210, 2007.
5. Medhi Kaytoue, Sergei O. Kuznetsov, Amedeo Napoli, Sebastien Duplessis, Mining gene expression data with pattern structures in formal concept analysis, *Information Sciences*, Volume 181, Issue 10, 15 May 2011, pp. 1989-2001, Special Issue on Information Engineering Application Based on Lattices, Elsevier, New York, 2011.
6. Nizar Messai, Marie-Dominique Devignes, Amedeo Napoli, Malika Smail-Tabbone, Querying a bioinformatic data sources registry with concept lattices, *ICCS'05 Proceedings of the 13th international conference on Conceptual Structures: common Semantics for Sharing Knowledge*, 323-336, 2005.
7. Medhi Kaytoue-Uberall, Sebastien Duplessis, Amedeo Napoli, Using Formal Concept Analysis for the Extraction of Group of Co-expressed Genes, *Modelling, Computation and Optimization in Information Systems and Management Sciences, Communications in Computer and Information Science* Volume 14, 439-449, 2008.
8. Benjamin J. Keller, Felix Eichinger, and Matthias Kretzler, Formal concept analysis of disease similarity, *AMIA Summits Translational Science Proceedings*, 42-51, 2012.
9. The Ensembl Genome Browser, http://www.ensembl.org/.